Good Afternoon everyone, my name is Arjun and I'll be presenting on the topic-"Neurological Disease Probability Prediction Using Voice Characteristics". My supervisor is Smt. M. Anila and my project ID is P36.

This is the agenda of my presentation.

Introduction:

As you may know, there are many neurological deceases known, such as Alzheimer's Disease or Parkinson's Disease. Around 1 billion people in the world suffer from a neurological disease. This project will be focusing on Parkinson's Disease. Almost every PD patient has their own set of symptoms which differ greatly from others, but a few basic symptoms are common in mostly all of them, namely resting tremors, stiffness, and slowness of movement. These resting tremors and stiffness contribute to another common symptom that is voice tremors or variation in speech. This variation in speech is the basis for my project.

This project aims to construct a model to employ vocal characteristics to classify and predict the probability of a person having a neurological disease like PD. Actually, there is no clinical test present at the moment to properly classify a person as a PD patient. So, this system aims to assist the doctors with diagnosis.

My first base paper has the title – "Automatically Detecting Errors and Disfluencies in Read Speech to Predict Cognitive Impairment in People with Parkinson's Disease". So, the mechanism of their model is that they used a manual transcript of sentences and ask the patient to try and read it. They record the patient and use that recording to generate an ASR based transcript, ASR stands for Automatic Speech Recognition. Then, they compare the two transcripts and evaluate the number of errors and disfluencies. They use a linear regression model to predict MoCA scores, MoCA stands for Montreal Cognitive Assessment. Basically, their model is built on the fact that patients with PD suffer from Cognitive Impairment. So, MoCA scores tell us if a person has CI or not. The total possible score is 30 points; a score of 26 or above is considered normal. Their dataset is a collection of audio recordings and transcripts from 37 people with PD who have a PD diagnosis.

My second base paper is named – "The Detection of Parkinson's Disease From Speech Using Voice Source Information". In this paper, they have split the study into two classifier approaches. The traditional approach and the end-

to-end approach. In the traditional approach, they use glottal features as speech signal. These features include aspects like noise, jitter, shimmer, etc. Then, they perform feature extraction and implement a SVM classifier, SVM stands for Support Vector Machine. This approach outputs a value to showcase whether a person has PD or a healthy control. In the end-to-end approach, they use the waveforms of the vocal features like frequency as input to a CNN model. In this, Max Pooling is done and the output of the Max Pool is given as input to a multi-layer perceptron which is nothing but an ANN with multiple fully connected layers. Max Pooling is a down-sampling strategy in CNN where on applying the filter, the maximum value is selected as output. The output of this model is a label predicting whether a person has PD or a healthy control. The database that they used is PC-GITA Speech Dataset.

The third base paper is named – "Advances in Parkinson's Disease detection and assessment using voice and speech: A review of the articulatory and phonatory aspects". This is an analysis paper studying the articulatory and phonatory features relevance in PD prediction systems. They concluded that these features are a great fit for the detection systems using vocal features.

In my proposed system, the dataset is a PD voice characteristics dataset whose vocal characteristics are extracted using a speech analysis software called Praat. I'll be using a Recurrent Neural network utilizing vocal features. The recurrent neural network will be implemented using the keras python library.

My approach will include pre-processing the data to reduce noise as vocal feature datasets tend to have a lot of noise. Then, I'll perform feature extraction using t-SNE model, it stands for t-distributed Stochastic Neighbor Extraction Model. I chose this feature extraction model as it has proven to be effective in high dimensionality datasets and provides a high accuracy for classification. Then comes the model training part, I'll be using 3 hidden layers in my model. Basically, at each time step, the input to a node will go through the activation function and that value will be passed to the hidden layer. Then, the hyperparameter tuning step, where I'll be experimenting with the parameters like number of epochs, number of nodes per layer, batch size, etc. If the data overfits, I may use the dropout concept where in, you reduce the percentage of training data given to the model. Then, I'll evaluate my model using the evaluation metrics like accuracy, precision, etc.

These are my references.

Thank You.